



המכללה האקדמית של תל-אביב-יפו
121113 – ארגון המחשב ושפת-סף
אביב תשס"ב

מבחן סופי – מועד א'

פתרון

משך הבחינה: 3 שעות.

חומר עזר מותר: מצורפים דפי עזר (השימוש בדפי העזר – באחריות הסטודנטים בלבד). מותר שימוש במחשבוני (אך לא במחשב אישי).

מרצים: דורון שומרון, אליאב גנסין

ניקוד: המבחן מחולק לשני חלקים

חלק א' – עליך לבחור 4 מתוך 6 שאלות, כל אחת 15 נקודות – סה"כ 60 נקודות
חלק ב' – 2 שאלות חובה, כל אחת 20 נקודות – סה"כ 40 נקודות

במבחן זה 11 עמודים, כולל דף זה, לא כולל דפי העזר.

בהצלחה !

שם הסטודנט: _____

תעודת זהות/מס' סטודנט: _____

חלק א'

1. (15 נקודות)

נתון כי ערכו של רגיסטר DS הוא 1000h.

נתונות השורות הבאות מתוך תכנית אסמבלר:

```
.data
VAR1=10
VAR2 DW 4 DUP(?,?),3
VAR3=15
VAR4 DW VAR1
VAR5 DD VAR4
VAR6 DB VAR3
```

א. המשתנה VAR6 יימצא בכתובת פיזית **10018h**

ב. לאיזה ערך יאותחל המשתנה VAR6? (כתוב/כתבי בבסיס הקסדצימלי)

0Fh

ג. לאיזה ערך יאותחל המשתנה VAR5? (כתוב/כתבי בבסיס הקסדצימלי)

00000012h

מי שרשם אחת מהתשובות – 00010012h, ???0012h - התקבלה תשובתו גם כן (למרות שאינה

תשובה מדויקת)

2. (15 נקודות)

התכנית הבאה היא תכנית לבדיקת פולינדרום.

פולינדרום היא מחרוזת שנראית אותו דבר כשהיא נכתבת הפוך.

התכנית מדפיסה "y" אם המחרוזת היא פולינדרום, או "n" אחרת.

מלא/י את השורות החסרות.

```
code segment
    mov     bp,array
    mov     bx,array
    mov     ah,0 ; /* (א) */
input:    int     16h
    cmp     al,10d ; read string until newline
    je     testpal
    mov     [bp],al ; /* (ב) */
    inc     bp
    jmp     input
testpal:
    dec     bp
    mov     al,'n'
begloop:  cmp     bx,bp
    jae     ispal ; it is palindrome
    mov     cl,[bp] ; /* (ג) */
    cmp     cl,[bx] ; /* (ד) */
    jne     notpal ; not a palindrome
    inc     bx ; /* (ה) */
    dec     bp
    jmp     begloop ; /* (ו) (ז) */
ispal:   mov     al,'y' ; /* (ח) */
notpal:  mov     ah,14 ; /* (ט) */
    int     10h ; /* (י) */
finish:  mov     ax,4c00h
    int     21h
array:   db     ?
code ends
```

3. (15 נקודות)

נתון קטע הקוד הבא:

```

code segment
    mov     [mybyte], 5 ; /* (א) */
    mov     sp, 0468h
    xor     ax, ax
here:     add     al, [mybyte]
          push  ax
          dec   [BYTE mybyte]
          jnz  here
          pop  es
          nop

```

א. מלא/י את החסר בתכנית שלמעלה כך שכשנגיע לפקודה nop ערכו של sp יהיה 0460h. מה יהיה ערכו של AX כשנגיע לפקודה nop ?

הערך החסר 5

ערכו של AX 0Fh

ב. מלא/י את הטבלה הבאה, כך שתייצג את תוכן הזיכרון כאשר אנו מגיעים לפקודה nop:

Address	Contents
SS:0467h	<u>00h</u>
SS:0466h	<u>05h</u>
SS:0465h	<u>00h</u>
SS:0464h	<u>09h</u>
SS:0463h	<u>00h</u>
SS:0462h	<u>0Ch</u>
SS:0461h	<u>00h</u>
SS:0460h	<u>0Eh</u>
SS:045Fh	<u>??</u>
SS:045Eh	<u>??</u>

4. (15 נקודות)

נתונה הגדרת המאקרו הבאה:

```

PRT_CHAR MACRO      A_CHAR
    PUSH        AX
    IFDIF      <A_CHAR>, <AL>
        IFDIF   <A_CHAR>, <AX>
            MOV  AL, A_CHAR
        ENDIF
    ENDIF
    MOV        AH, 14
    INT        10h
    POP        AX
ENDM

```

א. מה עושה המאקרו PRT_CHAR?

המאקרו מדפיס את הפרמטר A_CHAR במידה והוא תו, או מדפיס את רגיסטר AX או AL אם שמו הועבר כפרמטר.

ב. איך נראית תוצאת הקריאה PRT_CHAR AL?

PUSH AX _____

MOV AH,14 _____

INT 10h _____

POP AX _____

5. (15 נקודות)

א. כתוב/כתבי סדרות פקודות שתגרומנה לכך שבמהלך הרצת התוכנית:

- בכל פעם שנלחץ המקש CTRL+BREAK יבוצע PRINT SCREEN
- בכל פעם שנלחץ המקש PRINT SCREEN יבוצע CTRL+BREAK

```

pushf
push ax
mov ax,3505h
int 21
mov dx,bx
push es
pop ds
mov ax,3523h
int 21
mov ax,2523h
int 21
mov dx,bx
push es
pop ds
mov ax,2505h
int 21
pop ax
popf

```

הערות:

- א. מותר להשתמש במחסנית.
- ב. וקטורי הפסיקות של PRINT SCREEN ו CTRL+BREAK הם 05h ו 23h בהתאמה.

ב. הסבר/הסבירי איזו פעולות נדרשות על מנת לגרום לכך שהשינוי שביצעת בסעיף א' יישמר גם לאחר סיום התכנית שלך. נמק! לא נדרשות פעולות נוספות מאחר ואין צורך בשמירת זכרון לפונקציית TSR כי הפונקציות היו קיימות בזכרון גם לפני כן.

6. (15 נקודות)

נתונה התכנית הבאה:

```
mmm macro x,y
    if x lt 0
        y=0-(x)
    else
        y=x
    endif
endm

cnt macro x1,x2
    z=x1
    mmm <x2-x1>,w
    rept (w+1)
        db %z
        z=z+2
    endm
endm

data segment
    n equ 3
    k equ 7
    cnt 5,9
    cnt n,k
    cnt k,n
data ends
```

הראה איך נראית הפרישה של ה data segment לאחר הרצת תכנית זו.

חלק ב'

7. (20 נקודות)

נתונה רשימה מקושרת לא ממוינת של ציוני תלמידים במעבדה באסמבלר. מזכירות התלמידים מעונינת למיין את הרשימה על פי מספר התלמיד.

לכל תלמיד רשומה בת 4 בתים הכוללים:

סיביות 11-0: מספר התלמיד.

סיביות 15-12: קוד המחשב שעליו עובד התלמיד במעבדה.

סיביות 31-16: כתובת רשומת התלמיד הבאה.

בהנחה שאזור הנתונים הוגדר מראש החל מכתובת MABADA ושהרשומה האחרונה מצביעה לכתובת המשך 0FFFFh, כתוב/כתבי את קטע הקוד (code segment) של תכנית המסדרת את הרשימה על פי הדרישה של מזכירות התלמידים.

להלן דיאגרמה של מבנה השדות עבור תלמיד אחד:

מספר התלמיד	קוד המחשב	כתובת רשומת התלמיד הבאה
0	11 12 15 16	31

.8 (20 נקודות)

נתונה התכנית הבאה:

```

push_regs macro      r1,r2,r3,r4,r5,r6,r7,r8,r9
    irp              reg,<r9,r8,r7,r6,r5,r4,r3,r2,r1>
        ifnb        <reg>
            push    reg
        endif
    endm

endm

pop_regs  macro      r1,r2,r3,r4,r5,r6,r7,r8,r9
    irp              reg,<r1,r2,r3,r4,r5,r6,r7,r8,r9>
        ifnb        <reg>
            pop     reg
        endif
    endm

endm

data      segment
matrix    dw 1,2,-4,5
          dw 3,-4,-5,7
          dw -6,2,-8,9
          dw 9,4,6,-5
n         dw 4
res       dw ?
result    dw ?
data      ends

stack     segment stack
          dw 50 dup(?)
top       label word
stack     ends

```

המשך התכנית בעמוד הבא

```

code      segment
          assume cs:code,ds:data,ss:stack
fij       proc
          push      bp
          mov       bp,sp
          push_regs ax,bx,cx,dx
          mov       ax,[bp+10]
          mov       bx,[bp+8]
          mov       cx,[bp+6]
          dec       ax
          shl       ax,1
          mul       cx
          dec       bx
          add       bx,bx
          add       bx,ax
          add       bx,offset matrix
          mov       ax,[bx]
          mov       bx,[bp+4]
          mov       [bx],ax
          pop_regs  ax,bx,cx,dx
          pop       bp
          ret       8
fij       endp
start:    mov       ax,data
          mov       ds,ax
          mov       ax,stack
          mov       ss,ax
          lea      sp,top
          mov       ax,0
          mov       bx,2
          mov       dx,n
          mov       si,offset res
again1:   push      dx
again:    push      bx
          push      dx
          push      n
          push      si
          call     fij
          cmp      res,0
          jl       hem
          inc      ax
hem:      inc      dx
          cmp      dx,n
          jbe     again
          pop      dx
          dec      dx
          inc      bx
          cmp      bx,n
          jbe     again1
          mov     result,ax
          mov     ax,4c00h
          int     21h

```

א. (3 נקודות) בתכנית מוגדרת מטריצה. צייר את המטריצה (בהצגה מתימטית)

ב. (10 נקודות) הרץ את התכנית. מה מוחזר ב result? (עשרונית)

ג. (3 נקודות) הגדר את הפרוצדורה `fij`. מה הפרמטרים שלה ומה היא מבצעת?

ד. (4 נקודות) מה מבצעת התכנית? (ניתן להמחיש באמצעות ציור על המטריצה)
